An Adaptive Cross-Correlation Algorithm for Extended-Scene Shack-Hartmann Wavefront Sensing

Erkin Sidick

Joseph J. Green, Catherine M. Ohara, and David C. Redding

Jet Propulsion Laboratory

California Institute of Technology

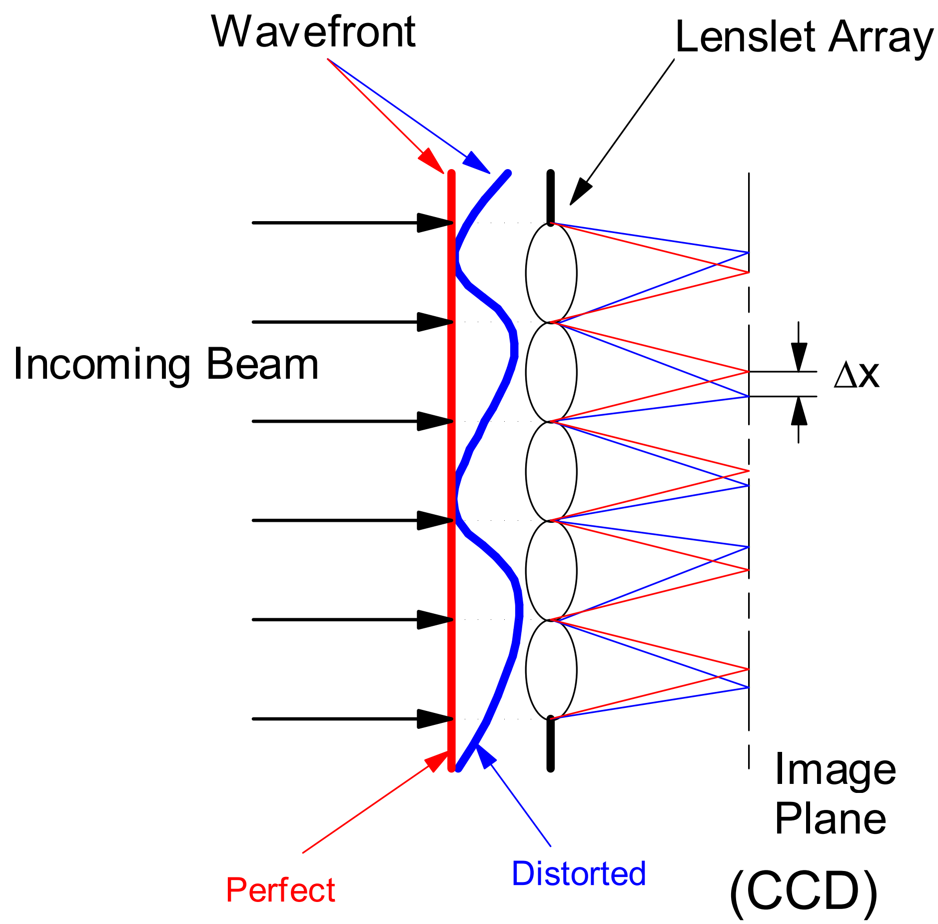
SRS 2007 in Vancouver, BC, Canada, June 18-20, 2007. Paper JTuA

National Aeronautics and
Space Administration

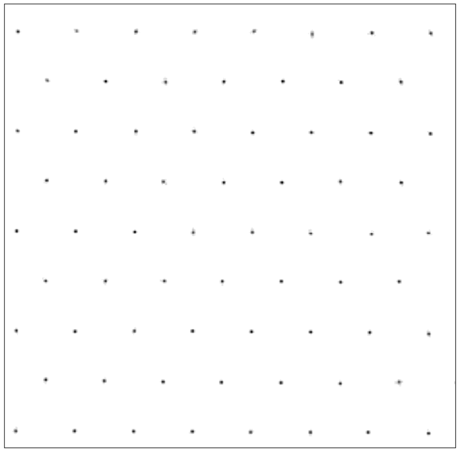
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

Principle of Conventional SH-WFS

- A Shack-Hartmann sensor places a lenslet array at a plane conjugate to the WF error source
- Each sub-aperture lenslet samples the WF in the corresponding patch of the WF
- When observing a star, the image is an array of spots, each of which is a sub-aperture PSF
  - Δx is proportional to local wavefront tilt
  - Wavefront-sensing → Finding Δx for all sub-images
  - Use centroiding (center-of-mass) method to find Δx

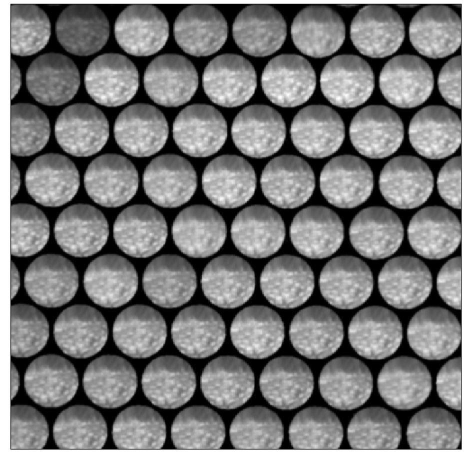


Extended-Scene S–H WFS

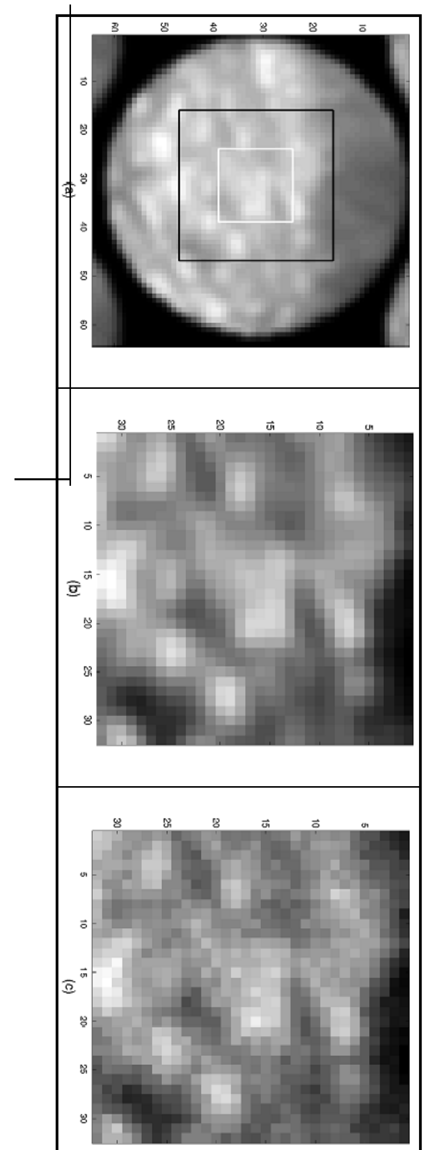


(a)

(b)



(c)

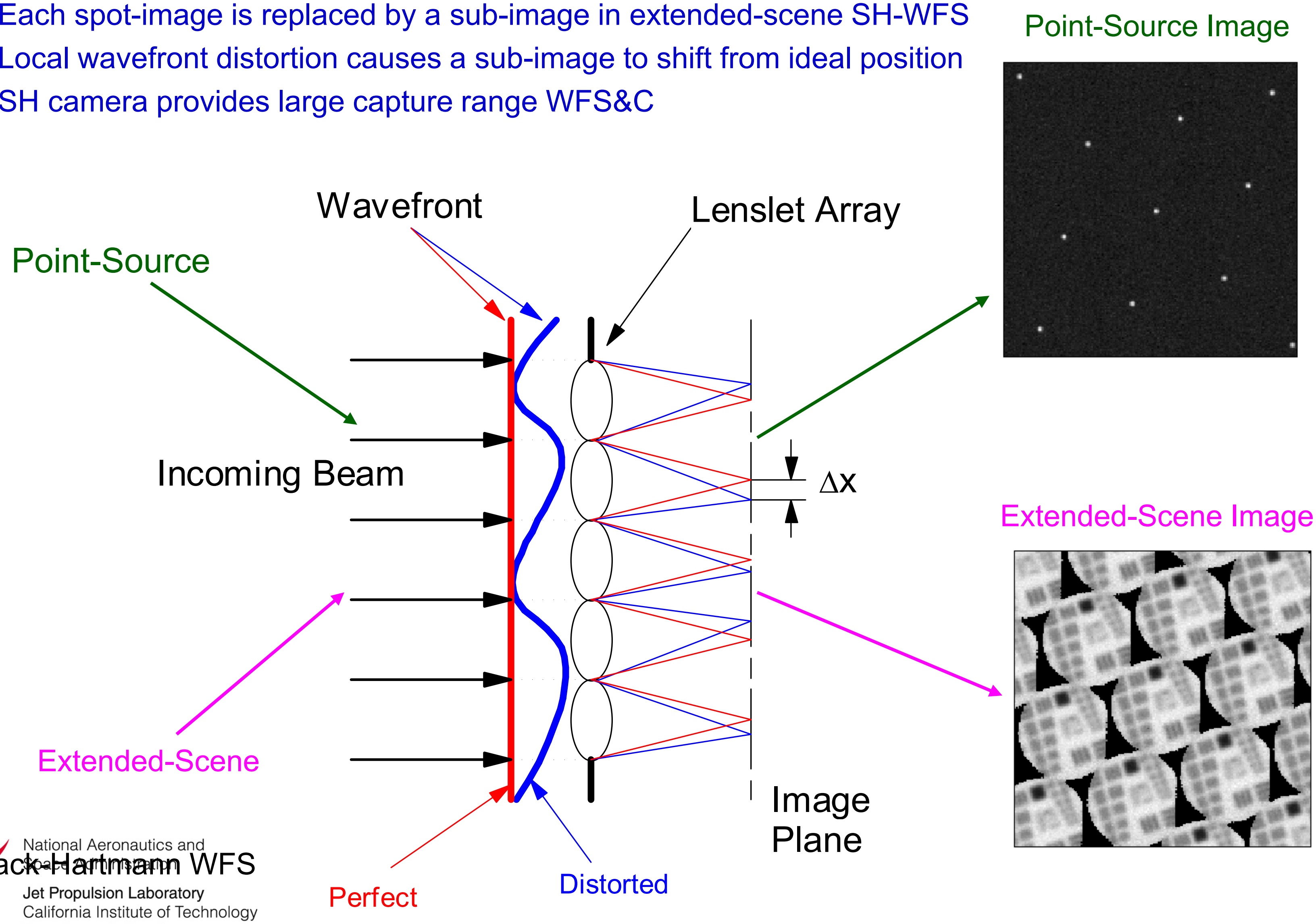


- The Shack-Hartmann Camera produces images as the convolution of the conventional image (limited by a tight field stop) with a regular grid of subaperture PSFs, as above
- Each subaperture is much blurrier than the main image, as its diffraction limit is defined by the subaperture, not the full aperture

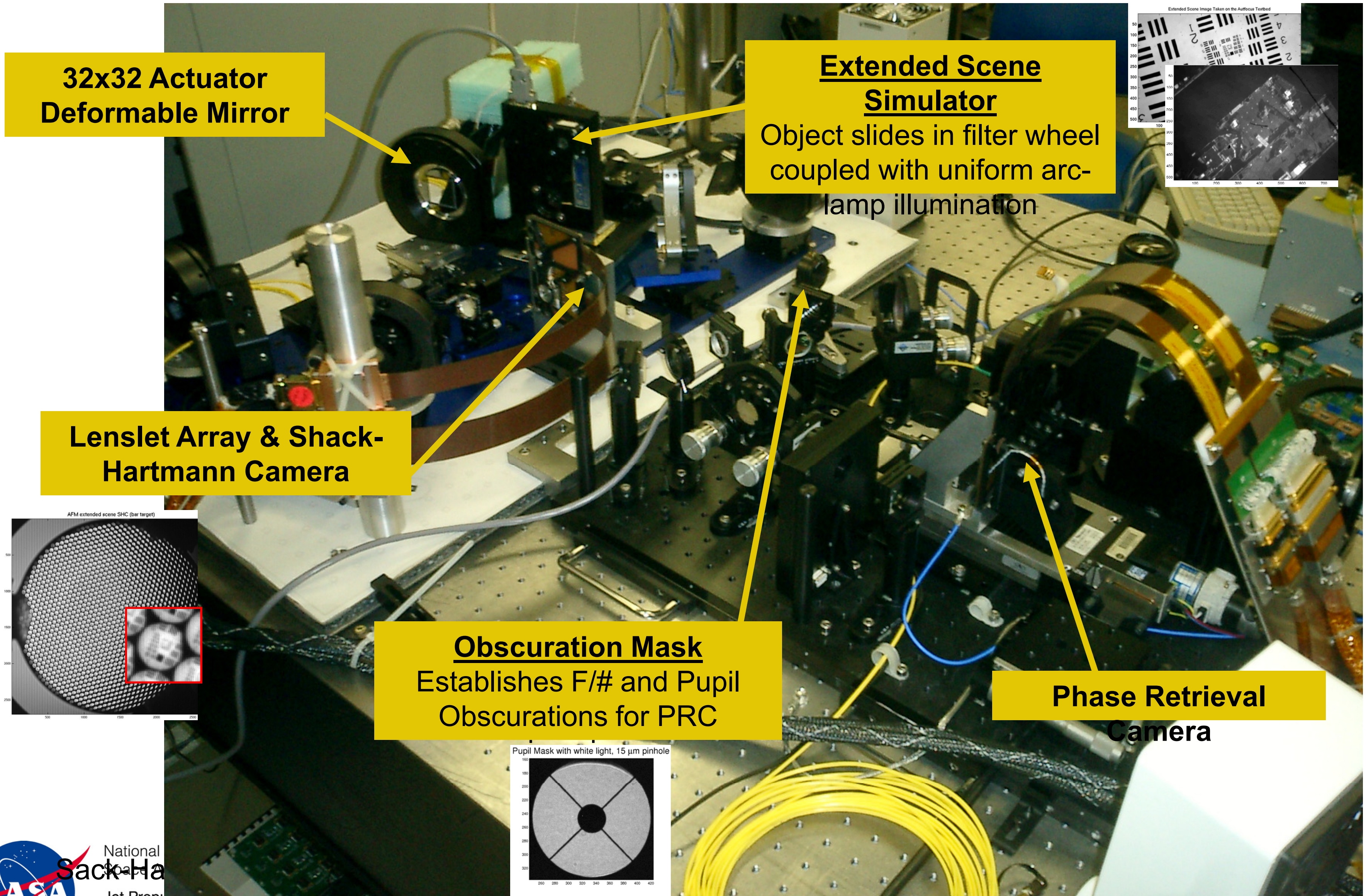
- Subaperture image shown at right at full size (64x64)
- ACC algorithm finds the central 32x32 box, and then identifies the multi-pixel shift of the features in the inner 16x16 cell with respect to a reference subaperture
- The subaperture-to-subaperture cell shifts give a measure of subaperture tilt

Point-Source (Star) versus Extended-Scene

- Following images were measured at JPL SH-WFS Testbed
  - Can be used with both point-source and extended-scene
  - Each spot-image is replaced by a sub-image in extended-scene SH-WFS
  - Local wavefront distortion causes a sub-image to shift from ideal position
  - SH camera provides large capture range WFS&C



Autofocus Testbed



Adaptive Cross-Correlation (ACC) Algorithm — How it Works

- Property of Fourier-transform:
  - Shift in time-domain ←→ Linear-phase in frequency-domain
  – In Fourier optics, $t \rightarrow (x,v)$ and $f \rightarrow (u,v)$

- Fourier-transform pair—Shown as one-dimensional for simplicity:

$$
\begin{aligned}
s(x) &\leftrightarrow \hat{s}(u) \\
s(x-\Delta x) &\leftrightarrow \hat{s}(u)\,\mathrm{e}^{-j2\pi\Delta x u}
\end{aligned}
$$

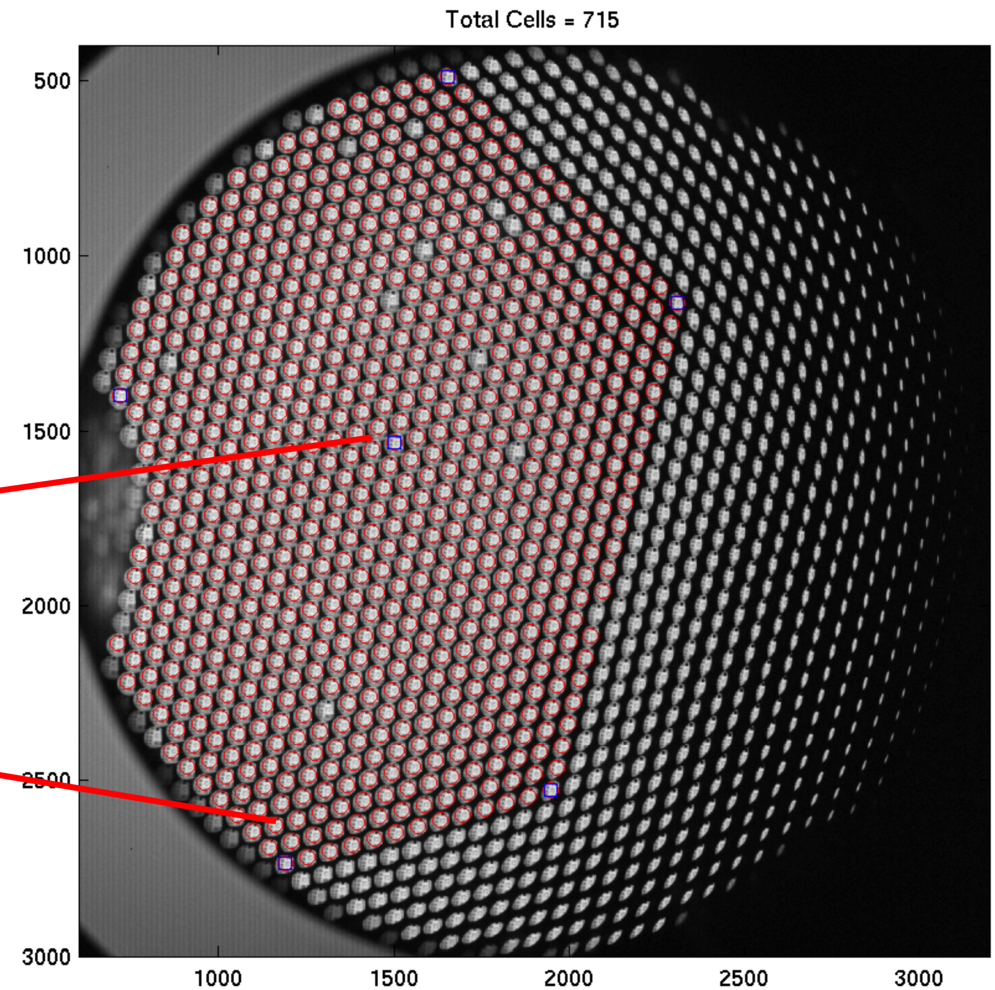
ACC Algorithm — How it Works (con.)

- In JPL testbed, only those cells marked with red-circle are used:
  - $r(x)$ = reference cell
  - $s(x)$ = test cells

- In ideal case:

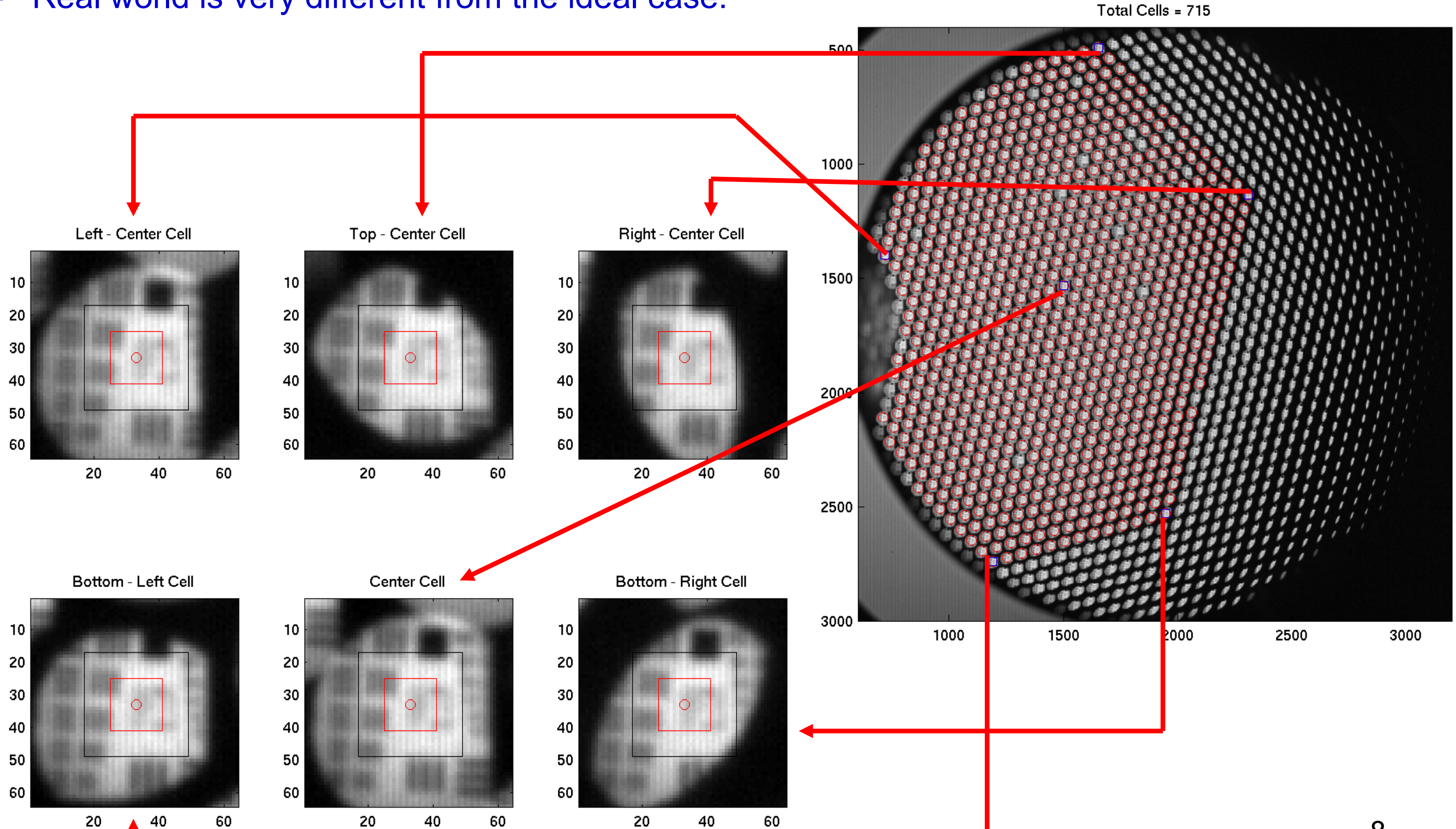
$$r(x) \leftrightarrow \hat{r}(u)$$

$$s(x) = r(x - \Delta x) \leftrightarrow \hat{r}(u)\,\mathrm{e}^{-j2\pi\Delta x u}$$



ACC Algorithm — How it Works (con.)

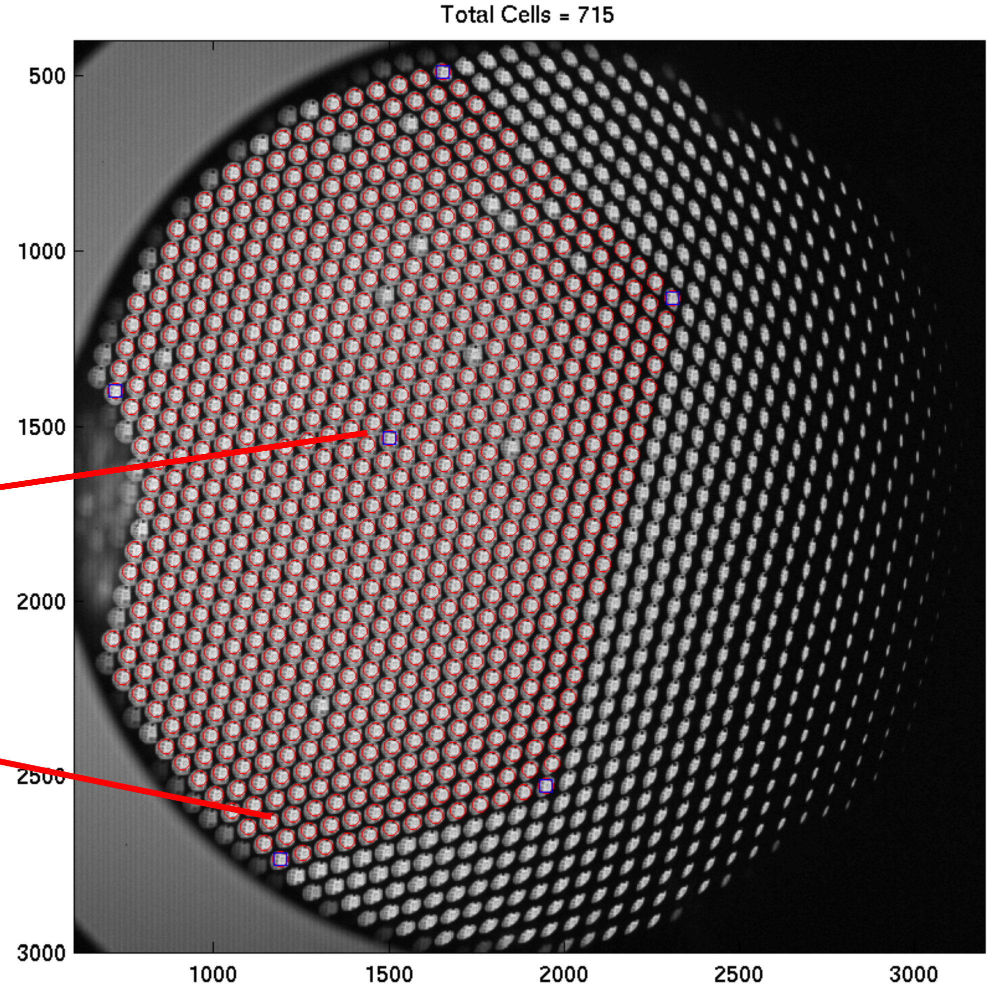
- Black square = usable sub-image (cell)
- Real world is very different from the ideal case:



Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

ACC Algorithm — How it Works (con.)

- Following illustration was made in one-dimension only. In reality, everything is 2-dimensional: $(x,y) \leftarrow\rightarrow (u,v)$



- In real world:

$$r(x) \leftrightarrow \hat{r}(u)$$

$$s(x) \neq r(x - \Delta x)$$

$$s(x) \leftrightarrow \hat{s}(u)$$

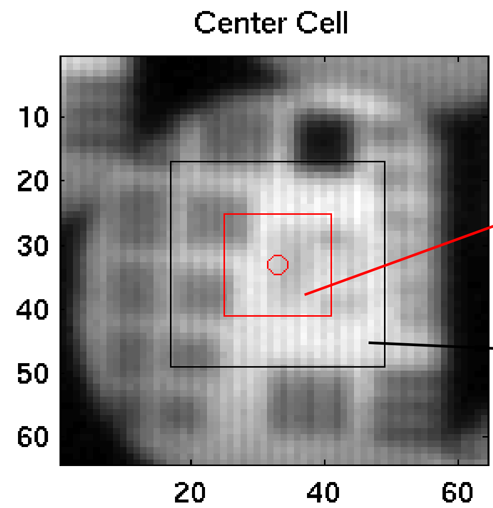
$$\hat{c}(u) = \hat{r}^*(u)\hat{s}(u) = \left|\hat{c}(u)\right| e^{j2\pi\varphi(u)}$$

$$\varphi(u) = \Delta x u + \varphi'(u)$$

Linear phase

Cross-correlation function

ACC Algorithm — How it Works (con.)



16x16 maxtrix $r(x,y)$ or $s(x,y)$:
Used to find $(\Delta x, \Delta y)$ from CC-function

32x32 maxtrix $S(x,y)$:
Used to shift $s(x,y)$ by $(-\Delta x, -\Delta y)$ to match it with $r(x,y)$

- Advantages of using smaller dimensions for $r(x,y)$ & $s(x,y)$:
  - Avoids wrap-around error when performing sub-image multi-pixels shifting
  - Makes the ACC calculations much faster
  - Increases the WFS dynamic range

- To shift $S(x,y)$ by $(-\Delta x,-\Delta y)$:
  - Obtain $S(u,v)$ by FFT $\rightarrow$ $S(u,v)\exp[-j2\pi(-\Delta xu-\Delta yv)]$ $\rightarrow$ (by IFFT) $S(x+\Delta x,\ y+\Delta y)$

ACC Flow-Chart

$\hat{r}(u,v) = FFT\{r(x,y)\}$

$i \le N_c$

No → STOP

Yes →

$\hat{S}_{i0}(u,v) = FFT\{S_i(x,y)\}$

$\Delta x = 0,\ \Delta y = 0$

$Counter = 0$

$Shift = \infty$

$Shift > Tol\ \&\ Counter < N_{iter}$

No → Store $\Delta x,\ \Delta y$

Yes →

$\hat{s}_i(u,v) = FFT\{s_i(x,y)\}$

$\hat{c}(u,v) = \hat{s}_i(u,v)\hat{r}^*(u,v) = |\hat{c}(u,v)|\exp[j2\pi\phi(u,v)]$

$[\delta x, \delta y] = Slope\{\phi(u,v)\}$

$Shift = \sqrt{\delta x^2 + \delta x^2},\quad Counter = Counter + 1$

$\Delta x = \Delta x + \delta x,\quad \Delta y = \Delta y + \delta y$

$\hat{S}_i(u,v) = \hat{S}_{i0}(u,v)\exp[-j2\pi(-\Delta x u - \Delta y v)]$

$S_i(x,y) = IFFT\{\hat{S}_i(u,v)\}$

$\hat{s}_i(u,v) = FFT\{s_i(x,y)\}:\quad 16 \times 16$ pixels

$\hat{S}_i(u,v) = FFT\{S_i(x,y)\}:\quad 32 \times 32$ pixels

For example:

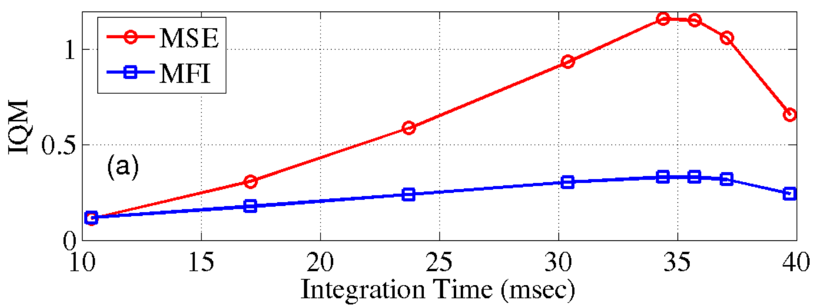
Tol = 0.01 pixels

N_{iter} = 15

32 x 32 Pixels Test Cells to be Analyzed

- Obtained 8 SH images with different integration time
- Used different cells for $r(x,y)$ & $s(x,y)$
- Red-square corresponds to a 16x16 pixels area

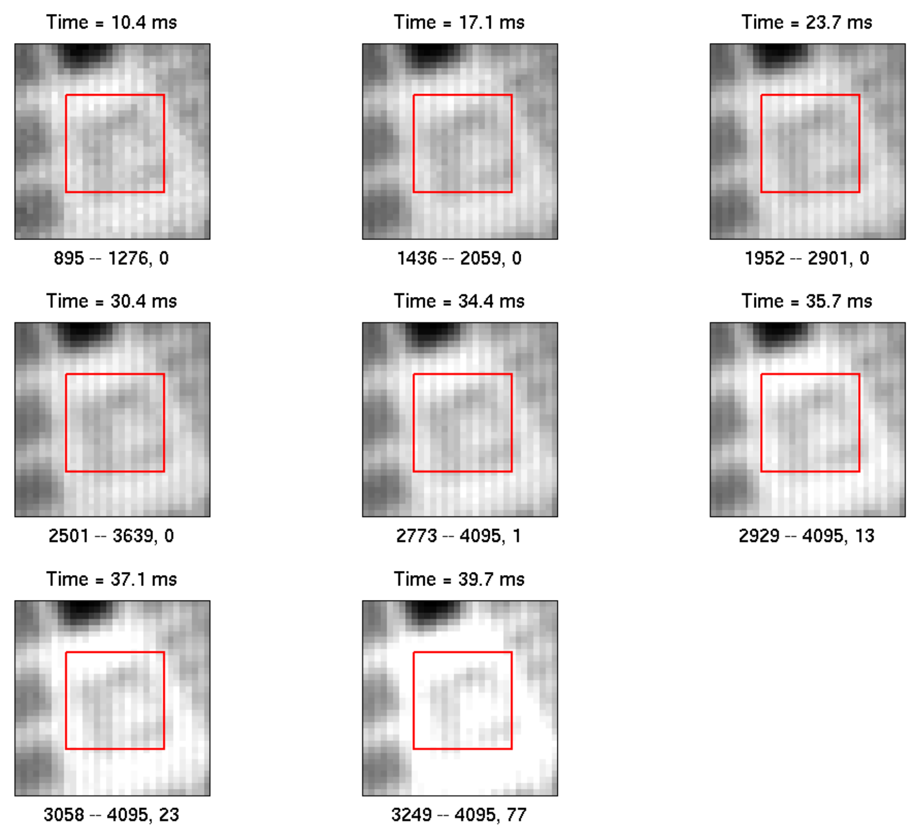
- Variations of IQM's with itegration time:
  - MSE = Mean-Squared Error
  - MFI = Modified Fisher-Information



$$MSE = \sum_{x,y} \left| g_i(x,y) - \bar{g}_i(x,y) \right|^2$$

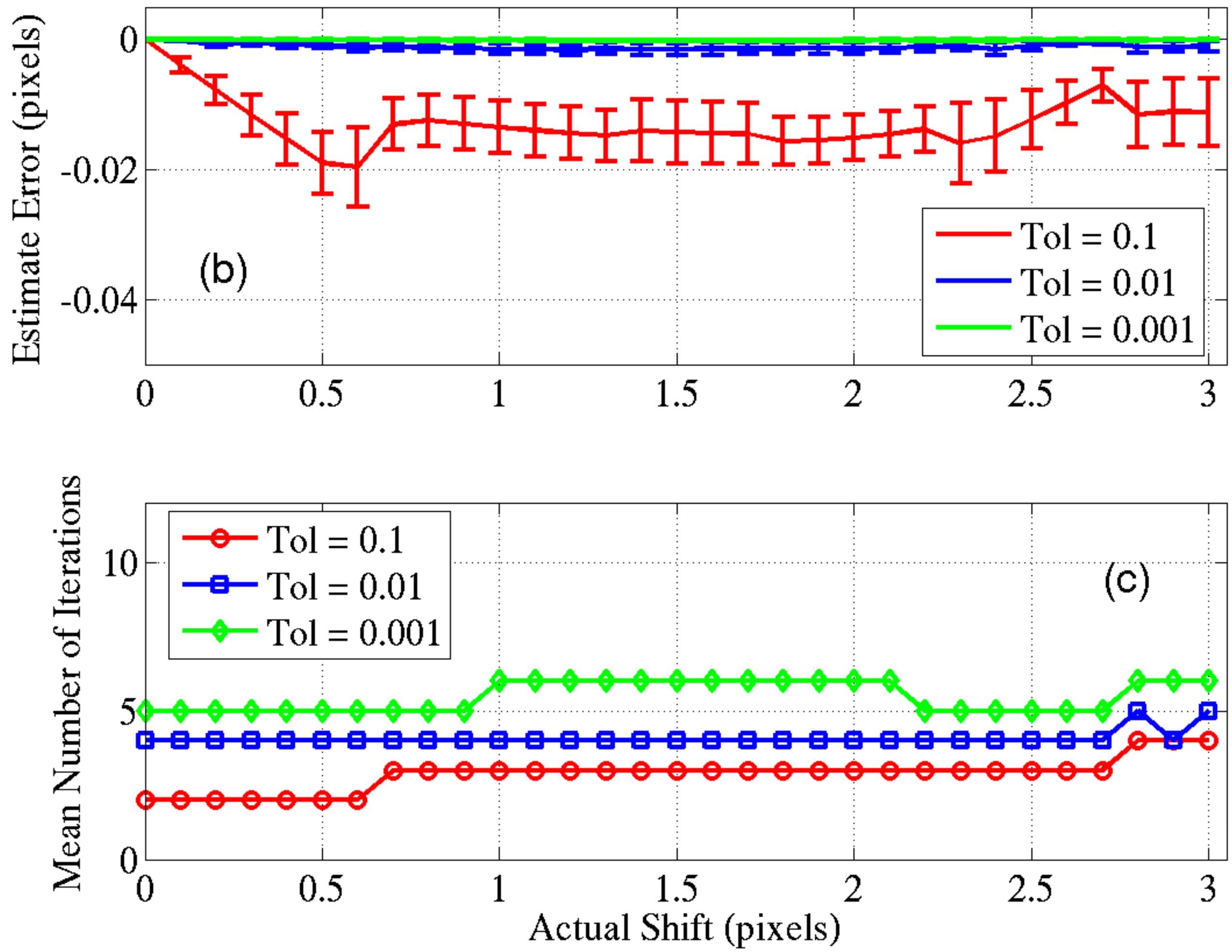
$$\Psi_i(x,y) = g_i(x,y) / 4095$$

$$MFI = 4 \times \sum_{x,y} \left[\nabla a_i(x,y) \right]^* \bullet \nabla a_i(x,y), \; a_i(x,y) = \sqrt{\Psi_i(x,y)}$$



Speed versus Tolerance

- Shifted $s(x,y)$ by known amount Δx, and determined the relative offset between $r(x,y)$ and $s(x,y)$ with ACC
- Used different cells for $r(x,y)$ & $s(x,y)$



National Aeronautics and Space Administration
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

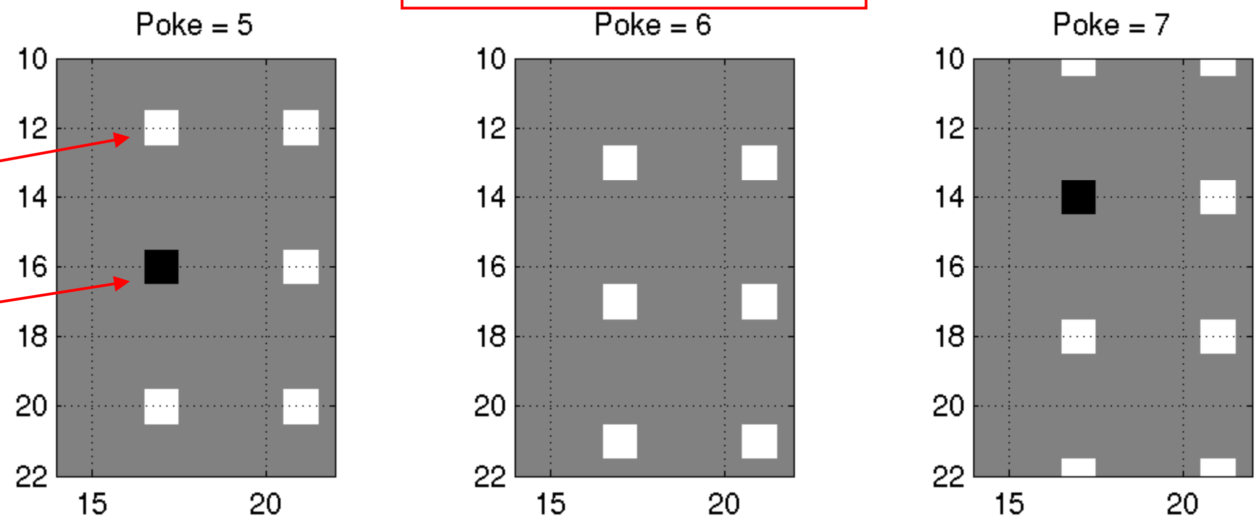
Sack-Hartmann

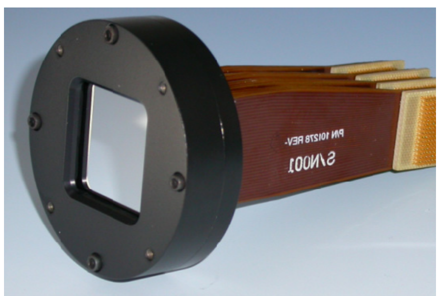
Examples: Point-Source Spot Image Analyzed with ACC

DM Poke Patterns

ΔV = +15V

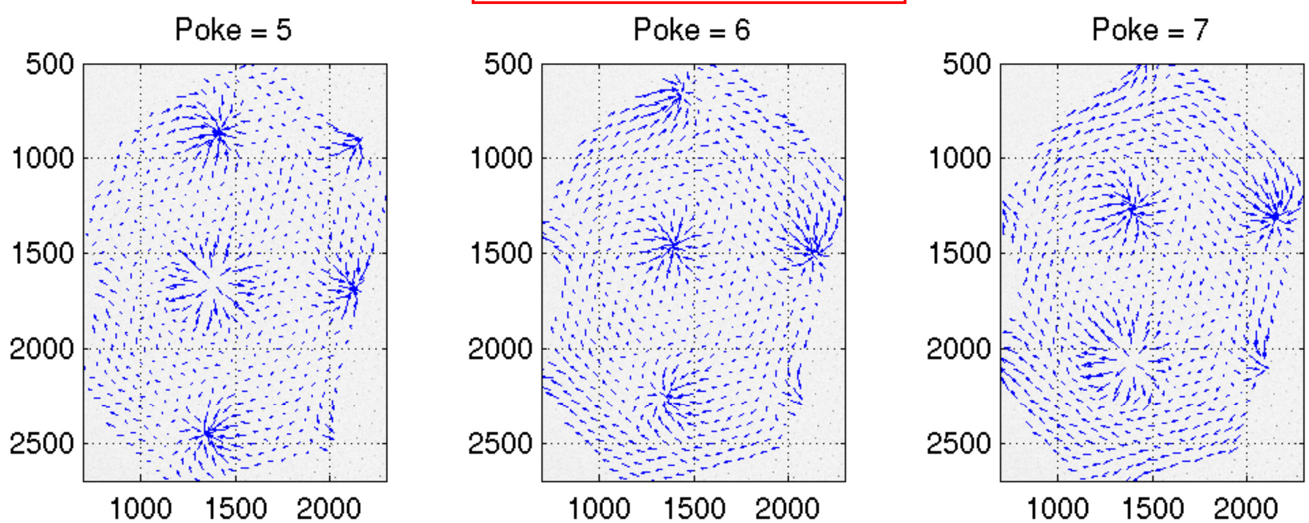
ΔV = -15V





Deformable-Mirror (DM)

Offset Diagrams



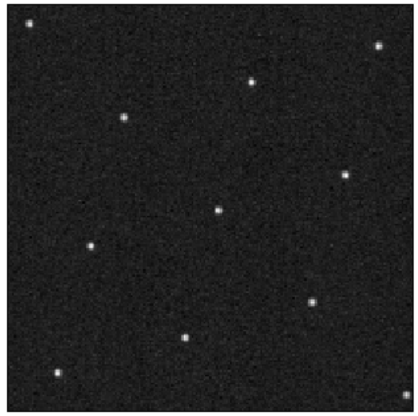
National Aeronautics and Space Administration

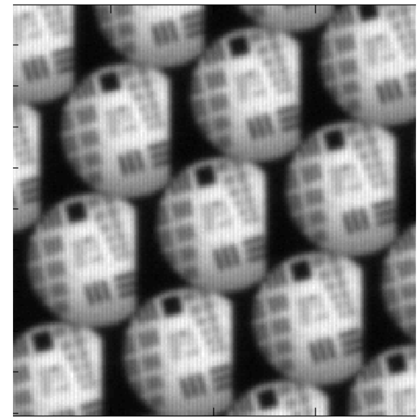
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

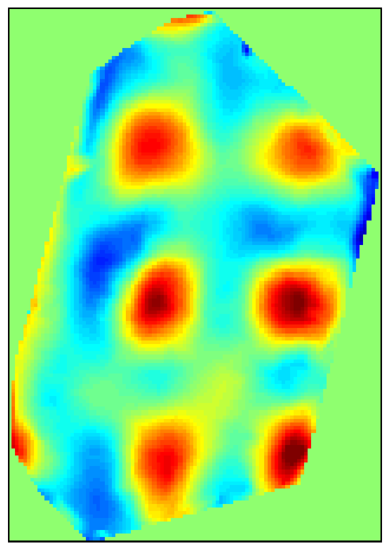
Extended Scene versus Point-Source

- Same poke patterns are used in both cases, but the measurement are done on different dates.

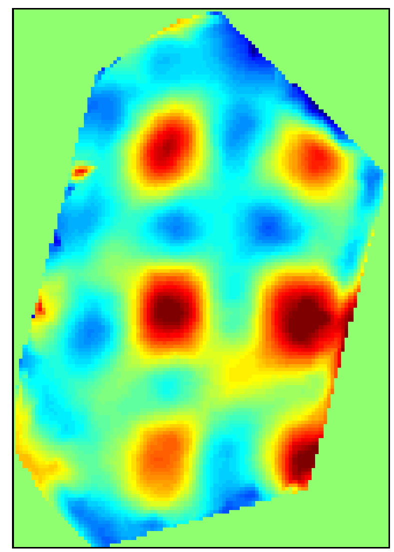
  - There are some differences in light path and actuator registration for point-source and extended scene, which is partially responsible for difference in OPD results.







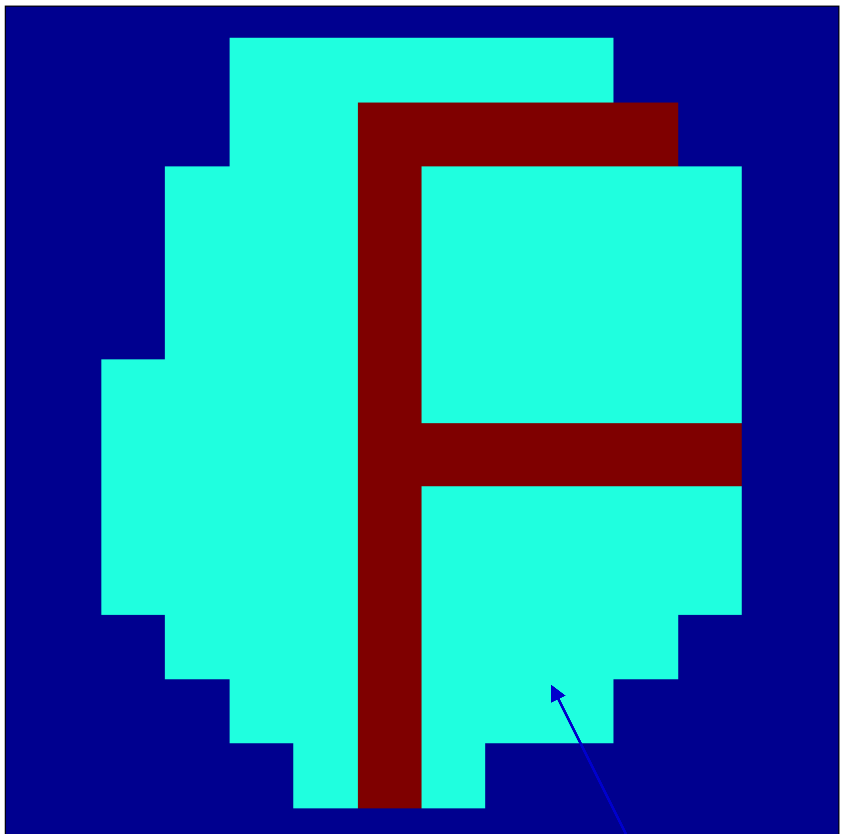
RMS = 50, PV = 299.5nm



RMS = 54.7, PV = 327.4nm

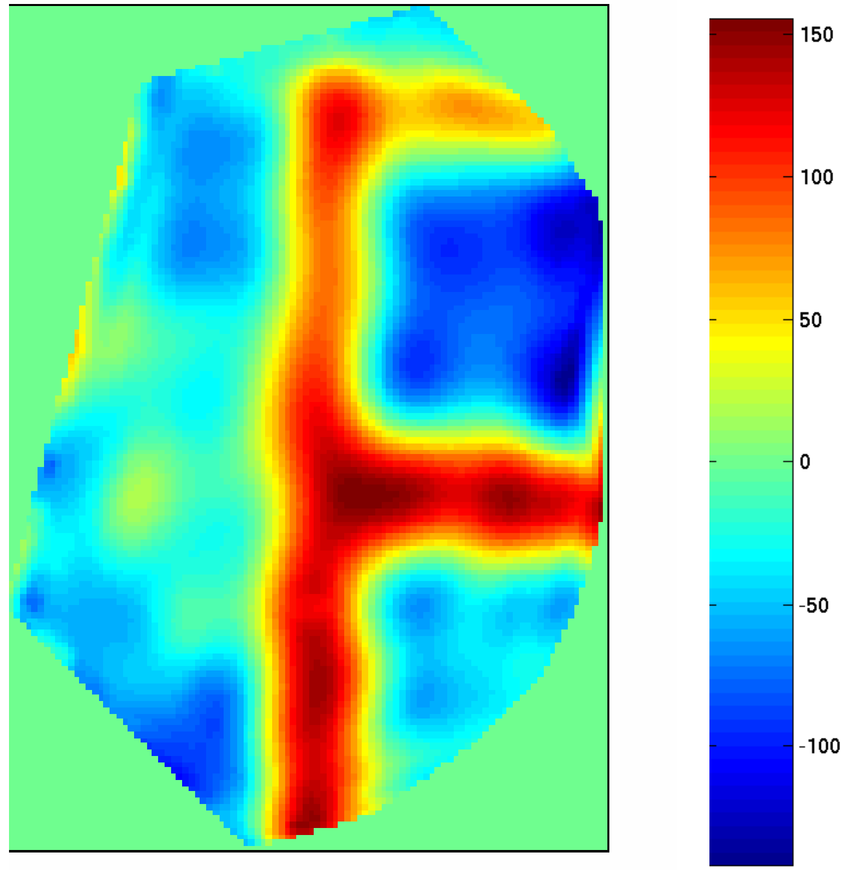
OPD Map Measured with Extended Scene

DM Poke Pattern



Active Window

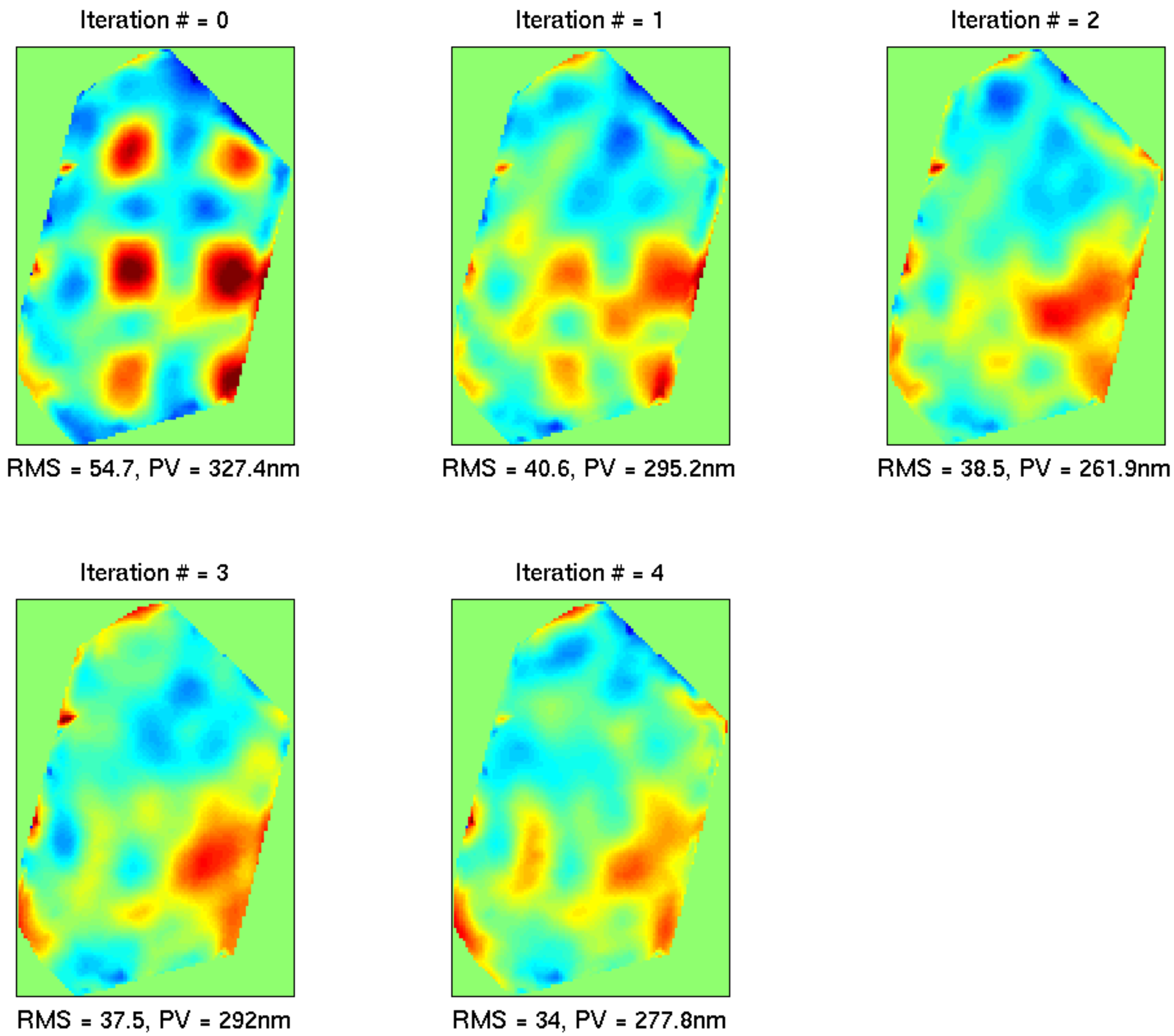
Reconstructed OPD Map



OPD is in nm

Example of Extended-Scene WFS&C

- There are 78 actuators inside the active window, but only 50 eigen-modes were used in this experiment.



Summary

- Extended-scene SH sensor is useful when point-source is not available but SH-WFS is needed
- ACC requires only about 4 image-shifting iterations to achieve 0.01 pixel accuracy
- ACC is insensitive to both background light and noise—much more robust than centroiding

- **Acknowledgement**
  – We thank Rhonda Morgan at JPL for her assistance with the S-H testbed